International Journal of Energy, Information and Communications
Vol. 2, Issue 3, August 2011

UnHOS: A Method for Uncertainty Handling in Commercial Off-The-Shelf (COTS) Selection

Hamdy Ibrahim[1], Abdel-Halim H. Elamy[2], Behrouz H. Far[1], Armin Eberlein[3]

[1]*University of Calgary, 2500 university Drive NW, Calgary, AB, T2N 1N4, Canada*
[2]*Department of Electrical and Computer Engineering, University of Alberta, Canada*
[3]*Department of Comp Science & Engineering, American University of Sharjah, UAE*
{hibrahi; far; eberlein}@ucalgary.ca and elamy@ualberta.ca

Abstract

Development of software systems using Commercial Off-The-Shelf (COTS) has been discussed over several years. Still, COTS-based development in particular the COTS selection process faces several challenges such as uncertainty related to COTS information and their vendors. COTS selection usually relies on information that may be uncertain in terms of completeness, accuracy, and consistency. Ignoring this uncertainty negatively influences the quality of COTS selection decisions and may lead to a low quality of the final software solution and ultimately stakeholder dissatisfaction. In this paper, we propose a method called UnHOS (Uncertainty Handling in COTS Selection) for evaluating COTS candidates while explicitly representing uncertainty. The UnHOS method uses the Analytic Hierarchy Process (AHP) to rank COTS candidates and the Bayesian Belief Network (BBN) to represent uncertainty. The final selection decision relies not only on the AHP ranking but also on the evaluation team's confidence level, created by BBN, about the AHP ranking. Furthermore, we present a software tool developed to support evaluators during the use of the UnHOS method and to improve its usability. We also present a case study of an airline reservations system to show how the UnHOS method and its tool can be used in practice.

Keywords: *COTS-Based Development, Uncertainty Management, Bayesian Belief Networks, Analytic Hierarchy Process.*

1. Introduction

In the last several years, software development has been influenced by commercial off-the-shelf (COTS) software products. The use of COTS products offers potential benefits such as: (1) Reduced development cost due to the mass production of reusable components; (2) Reduced development time and effort as COTS products are ready-made; (3) Improved quality of the final software system as COTS products have previously been used and tested by other customers; (4) Rich functionality as COTS components have been developed for a wide spectrum of customers with different needs [1, 2]. The COTS-based software development (CBSD) process has five main phases (see Figure 1). Details about these phases can be found in [3-5].

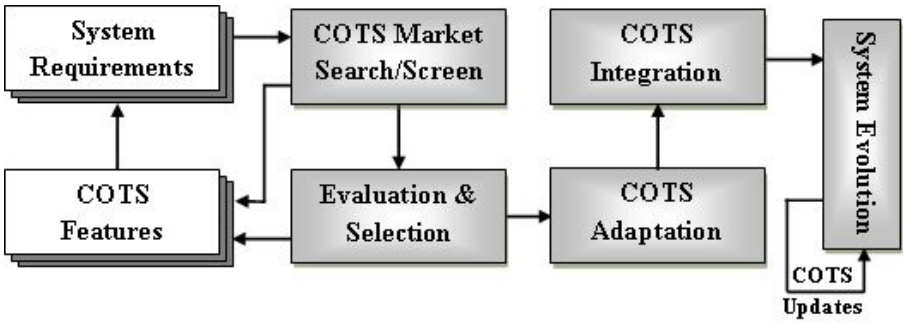

Figure 1: The CBSD Process

The COTS selection process plays a key role in the CBSD process and the success of the CBSD project. However, CBSD, in particular COTS selection, is a non-trivial task and associated with several challenges, such as uncertainty related to information about COTS products and their vendors [6-8]. Current COTS selection methods [9-21] assume that scores reflecting how much each candidate satisfies pre-defined criteria are certain. However, these scores are subject to uncertainty because the evaluation process is cut short due to limited resources in terms of time, budget, and workforce allocated to the CBSD project as well as the possibility of having incomplete, imprecise, and inconsistent information about the COTS products and their vendors.

In this paper, we propose a method, called UnHOS "Uncertainty Handling in COTS Selection", for evaluating COTS candidates and selecting the one that best fits stakeholder needs and project constraints while explicitly representing uncertainty. The UnHOS method uses the Analytic Hierarchy Process (AHP) to rank COTS candidates as well as addresses possible inconsistencies in candidates' information. The UnHOS also uses the Bayesian Belief Network (BBN) to represent uncertainty related to incomplete or inaccurate information about candidates. Therefore, UnHOS has two outcomes. The first outcome is the overall performance or score value created by AHP and represents how well each candidate performs based on a set of evaluation criteria. The second outcome is the confidence level created by BBN and represents the confidence of evaluators in the performance level of COTS candidates. Furthermore, we present a software tool that supports the UnHOS and improves its usability. The UnHOS and its supporting tool are used in this paper to select a COTS product for an airline reservations system and to identify the impact of uncertainty on the final COTS selection decision.

The rest of the paper is structured as follows: In Section 2, the research problem is defined and the research motivation is presented. Section 3 presents related work. This includes a discussion of a number of well-known COTS selection methods and how they address the uncertainty challenge. Section 4 gives an overview of the UnHOS and its supporting tool; Section 5 presents a case study in which a COTS product for an airline reservation system is selected. The case study demonstrates how UnHOS and its supporting tool are used. Finally, Section 6 summarizes our conclusions and future work.

2. Problem Definition and Research Motivation

Uncertainty has been defined as “A general concept that reflects our lack of sureness about something or someone, ranging from just short of complete sureness to an almost complete lack of conviction about an outcome” [22]. Uncertainty is a challenge that often impacts the quality of the COTS selection decision since making a good COTS selection

decision requires complete, consistent, and accurate information. Unfortunately, the information, on which decision makers rely to select a COTS product, is less than perfect. There may be missing, unknown, or ambiguous information that is of significant importance [23]. CBSD, in particular COTS selection, is subject to multiple sources of uncertainty. One potential source is the uncertainty related to COTS information [23]. Examples for this uncertainty include the following issues:

- The current state (e.g. which features exist), and the future state (e.g. which features will be added and/or removed) of a COTS product.
- The impact of the COTS product on the system under development.
- Hidden defects that still exist in COTS candidates.
- The compatibility of various COTS candidates to be integrated together to form a complete system.
- The possibility of tailoring or customizing an existing COTS candidate to better meet stakeholder needs and other project constraints and assumptions.
- The viability, financial stability, capability, reputation, support, and trustworthiness of COTS vendors.
- The evolution of the COTS market including new technologies and standards used during COTS product development and the cost of new releases.

As a result, it is difficult to assess the capabilities and weaknesses of COTS products with certainty because of the following reasons:

- No one has comprehensive knowledge about all the COTS software products and in most cases, the black-box nature of COTS products forces evaluators to rely on vendor claims.
- The description of COTS products is often incomplete.
- The evaluation process of COTS product features is usually carried out with limited resources (e.g. time, budget, workforce, etc.) and under assumptions that might not be applicable in real situation [23].
- The complexity of COTS products and rapid change in the COTS market are considered as inhibitors to gain high quality information about COTS candidates [25].

Having uncertain information about the problem context and COTS candidates may lead to a poor definition of evaluation criteria, and therefore in difficulties in discriminating between the candidates and ultimately the selection of an inappropriate candidate.

In summary, the evaluation team cannot be completely sure that assigned score values accurately reflect how well COTS candidates satisfy the evaluation criteria. Consequently, the evaluation process is subject to uncertainty which negatively influences the final selection decision and the likelihood of meeting project goals. The problem of selecting a COTS product considering uncertainty can be described as a 3-tuple (C_C, E_C, I_U) and represented by equation 1.

$$B_C = \sigma_{E_C}(C_C) \mid I_U \tag{1}$$

Where:

B_C: the best COTS candidate (B_C □ C_C).

C_C: the set of COTS candidates $\{C_i, i = 1,2,.....,l\}$

E_C: the set of evaluation criteria $\{E_j, j = 1,2,.....,m\}$. Each criterion is associated with a set of attributes A=$\{A_{jk}, j = 1,2,....,m \ \& \ k = 1,2,....,n\}$

I_U: the uncertainty associated with information. Uncertain information is characterized by the following attributes: (a) completeness representing whether there is missing, unknown, or ambiguous information that is of significant importance; (b) accuracy reflecting the correctness of information and (c) consistency representing possible conflict in information.

According to equation 1, the COTS selection problem is mainly represented as a selection of the best element (B_C) from the COTS candidates set C_C considering the satisfaction of a pre-defined set of evaluation criteria E_C and given that the information I_U about the candidates and their vendors are uncertain. Therefore, two key research questions arise:

1. ***To what extent does uncertainty influence the final COTS selection decision?***
2. ***How can the explicit representation of uncertainty be used to determine possible risks associated with the COTS selection decision?***

3. Related Work

3.1. COTS Selection Methods

Several COTS selection methods have been proposed [9-21]. The Off-the-shelf-options (OTSO) method consists of six main phases: search, screening, evaluation, analysis, deployment, and assessment. It provides techniques for defining evaluation criteria and comparing cost-benefits of COTS candidates. It also uses the analytic hierarchy process (AHP) to consolidate the evaluation results. It also discusses the importance of requirements in defining these criteria and the effect of evaluation criteria on the overall COTS selection process. It also realizes that the lack of attention given to requirements may lead to a suboptimal COTS selection [10]. Other similar multi-phase COTS selection methods include comparative evaluation process (CEP) [15], COTS acquisition process (CAP) [13], and PECA (Plan, Establish, Collect, and Analyze) [20]. CEP realizes the importance of tackling uncertainty by assigning a credibility level to the data sources used during the evaluation process [15]. CAP is a measurement-oriented method where the evaluation process is configured based on an estimation of the measurement effort [13]. PECA is a high-level process which can be tailored to fit project properties and constraints. It suggests the involvement of stakeholders in the evaluation process, the screening of COTS products, and the documentation of the evaluation process for purposes of knowledge collection. It also asserts the importance of the accuracy of data collection and relates it to the confidence in the final evaluation results [21]. Other methods such as the COTS-based requirements engineering (CRE) method [14], COTS-aware requirements engineering (CARE) [21], storyboard process [18], combined selection of COTS components (CSCC) [19], and procurement-oriented requirements engineering (PORE) [9] are requirements-oriented methods that highlight the importance of specifying requirements as desirable needs rather than strict constraints since it is likely not possible to specify strict requirements and then

hope to find a group of COTS products that meets those requirements. These methods also support the idea of acquiring and validating requirements simultaneously with evaluating COTS candidates. The PORE and CARE methods are the only methods supported by a prototype tool named PORE process advisor [9] and CARE assistant tool [21] respectively. Table 1 shows to what extent current COTS selection methods address the uncertainty challenge and whether these methods are supported by a tool or not.

Table 1: Comparison of COTS Selection Methods with respect to Uncertainty

Comparison Factor	COTS Selection Methods									
	OTSO	CEP	CAP	CRE	PORE	CARE	STACE	CSCC	PECA	Storyboard
Uncertainty	▼	●	▼	▼	▼	▼	▼	▼	●	▼
Tool Support	No	No	No	No	Yes	Yes	No	No	No	No
▼: Not Addressed									● : Partially Addressed	

According to Table 1, current COTS selection methods only partially or not at all address uncertainty. None of these selection methods consider uncertainty in a comprehensive manner. Only two methods (i.e. PORE and CARE) are supported by a prototype tool. The unavailability of a software tool supporting other COTS selection methods negatively influences their usability.

3.2. Current Trends to Address Uncertainty in CBSD

Uncertainty management is essential to improve the accuracy and quality of COTS selection decisions. Ignoring it may lead to sub-optimal selection, low quality of the final solution, and stakeholder dissatisfaction [26]. Several researchers have attempted to address uncertainty and manage its consequences [27-28, 23]. In [27], a Bayesian belief network is used to certify the reliability of COTS software systems. [28] discusses how uncertainty related to COTS license cost can be managed and how associated risks can be mitigated. [23] realizes the importance of doing COTS assessment while uncertainty is explicitly represented. It has adapted a model developed in [29] and used it for assessing a pair of COTS products considering the fault logs that might be available for the COTS candidates being assessed.

However, none of these attempts can be considered as a comprehensive solution that covers the COTS selection process taking into account various forms of uncertainty. This is because their focus is only on a specific evaluation perspective (e.g. reliability, license cost) not on the overall selection process.

4. UnHOS: A COTS Selection Method

UnHOS is a method (see Figure 2) for evaluating COTS candidates while considering uncertainty related to scores assigned to COTS candidates. UnHOS is considered as a hybrid method since it uses two techniques: AHP as an evaluation and ranking technique and BBN as an uncertainty representation technique. AHP was selected and implemented in the current

release of the supporting tool because it is a well-known decision making technique used by many existing COTS selection methods. In addition, AHP is based on pair-wise comparison of COTS candidates which is a more practical and accurate way than assigning an absolute value to each candidate. Besides, the AHP decision matrix contains redundancy thus increasing consistency and reducing possible judgment errors. AHP is used during the evaluation process to rank various candidates as well as to address possible inconsistency in COTS candidate information during uncertainty management.

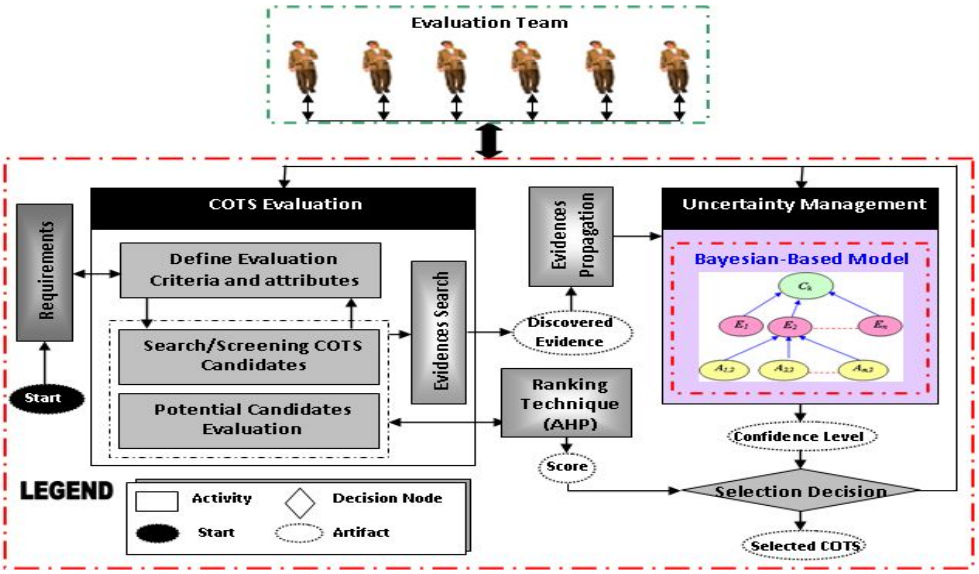

Figure 2: UnHOS Method

BBN was selected because it is a well-known technique for solving problems that involve reasoning under uncertainty and provides a graphically and mathematically sound technique for explicit representation of uncertainty. BBN mainly addresses the problem of having incomplete and inaccurate information about COTS candidates and vendors. The process of using the UnHOS method includes five main steps:

1. Eliciting an initial set of requirements from stakeholders. These requirements are used to define an initial set of criteria used to search the market to identify possible COTS candidates.
2. The features offered by these candidates are used to refine the criteria and the initial requirements set.
3. The refined criteria are used to filter COTS candidates and to select the most promising candidates. The process of searching for and screening of COTS candidates as well as refining system requirements and evaluation criteria continues simultaneously until reaching a reasonable number of potential and promising candidates.
4. Afterwards, potential candidates are evaluated in detail using AHP and uncertainty is simultaneously represented using BBN.
5. The results produced by AHP and BBN are used to select a candidate.

Assuming that a set of evaluation criteria are defined and the most promising COTS candidates are identified, the following subsections describe how AHP and BBN are used during the evaluation process. A brief description of the supporting tool is also provided.

4.1. Usage of AHP

The following subsections describe how AHP is used within the UnHOS method. The following three activities (See Figure 3) are performed.

4.1.1. Development of a Hierarchical Model

In the first step, a hierarchical model for the COTS selection problem is developed. The hierarchy model has three levels (see Figure 4). The first level is a goal level. In the COTS selection problem, the goal is to select the best COTS candidate from several candidates. The second level contains evaluation criteria against which candidates are evaluated. Finally, the alternatives level contains the candidates being evaluated.

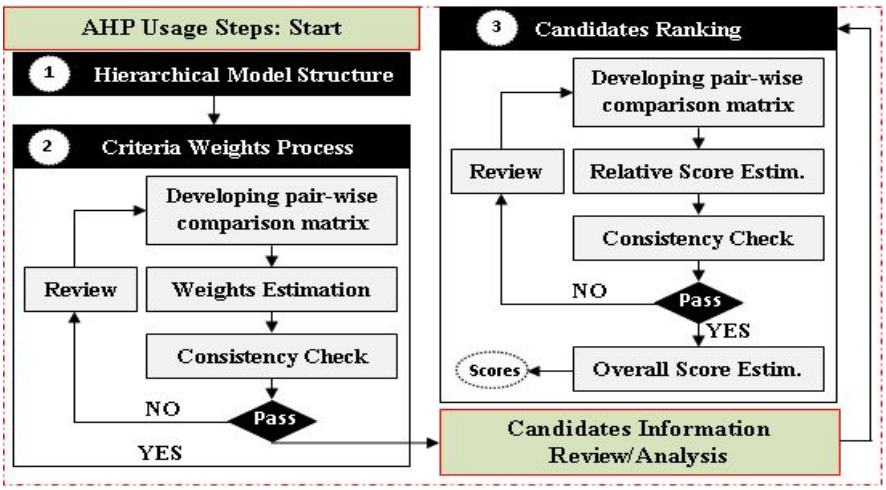

Figure 3: AHP Usage Process

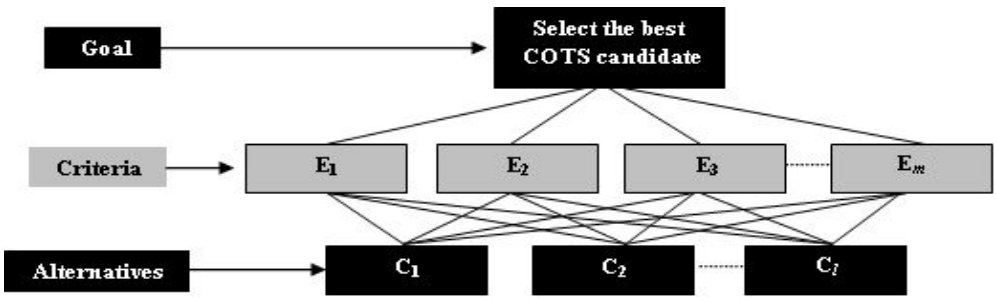

Figure 4: AHP Hierarchy for COTS Selection Problem

4.1.2. Criteria Weights Process

In this step, weights, namely (W_1, W_2,……., W_m) are calculated and assigned to the evaluation criteria. Weights reflect how important each criterion is to the evaluation team. Equation (2) must hold for all weights W_j.

$$\sum_{j=1}^{m} W_j = 1 \quad \textbf{(2)}$$

The estimation process starts with developing a pair-wise comparison matrix. The matrix cells represent the relative importance of each evaluation criterion compared to other criteria (see Table 2). Numbers from 1 to 9 are used to represent the relative importance **[32]**:

- 1 indicates that criteria E_1 and E_2 are of equal importance.
- 9 indicates that criterion E_1 is significantly more important than E_2.
- Values between 1 and 9 represent different levels of relative importance. While reciprocals indicate that E_2 is more important than E_1.

Table 2: Pair-wise Comparison Matrix for Criteria

Criteria	E_1	E_2	E_3
E_1	1	1/3	5
E_2	3	1	7
E_3	1/5	1/7	1

Then, equations (3-5) are used to convert comparison values into normalized rankings representing the weights of the compared criteria. A consistency check is also performed to make sure that the pair-wise judgments are consistent. If the judgments are inconsistent, the evaluation team will review the pair-wise comparison matrix and modify the values in the matrix to reduce the inconsistency.

Considering m criteria, the pair-wise comparison matrix, Comp_M, is a square matrix [mXm]. Equations (3-5) can be used to calculate the criteria weights. The following process is followed:

1. Calculating the sum of each column:

$$Column\_Sum[y] = \sum_{x=1}^{m} Comp\_M[x,y], for: y = 1,...,m \quad \textbf{(3)}$$

2. Normalization: Dividing each element in matrix Comp_M by the sum of its column.

$$Normalized\_M[x,y] = \frac{Comp\_M[x,y]}{Column\_Sum[y]}, where: x = 1,...,m \;\&\; y = 1,...,m \quad \textbf{(4)}$$

3. The criteria weight vector can be obtained by averaging the rows in the normalized matrix (i.e. Normalized_M[x,y]) as follows:

$$Weight\_Vector[x] = \frac{1}{m}\sum_{y=1}^{m} Normalized\_M[x,y], for: x = 1,...,m \quad \textbf{(5)}$$

Table 3 shows the criteria weights calculated for the pair-wise comparison matrix shown in Table 2.

Table 3: Criteria Weights

Criteria	Weight	Label
E_1	0.283	W_1
E_2	0.643	W_2
E_3	0.074	W_3

The consistency check is done by comparing a consistency ratio (CR) calculated by equation (6) to 0.1 [32]. A CR of 0 means that the pair-wise comparison values are perfectly consistent. Furthermore, when the CR is less than 0.1 [32], the comparison values are reasonably consistent. However, if the CR exceeds 0.1, the comparison values are inconsistent and evaluators need to review their judgments. More details about the calculation process of weights and the consistency ratio can be found in [32, 34-35].

$$CR = \frac{CI}{RI} \tag{6}$$

Where:

CI: consistency index for the pair-wise comparison matrix (i.e. *Comp_M*).

RI: An index [32] whose possible values are summarized in Table 4.

Table 4: Index Values [32]

No. of Candidates	1	2	3	4	5	6	7	8	9	10
RI	0	0	0.58	0.90	1.12	1.24	1.32	1.41	1.45	1.49

4.1.3. Candidates Ranking

The AHP technique is used for evaluating candidates and to rank them according their satisfaction of the evaluation criteria. The ranking process starts with a pair-wise comparison between COTS candidates with respect to all criteria (see Table 5). Each row of the comparison matrix contains a number of values representing a candidate's satisfaction with respect to a specific criterion and compared to other candidates. For example C_3 satisfies E_1 significantly more than C_4 does. Evaluators review and analyze information about the candidates to determine the pair-wise comparison values. Once the pair-wise comparison matrix is developed, it is used to calculate the relative score values, S_{ij} (i.e. relative score of candidate i with respect to criterion j) for the candidates (see Table 6). These relative score values represent the candidates' performance with respect to each individual criterion and compared to other candidates. After that, a consistency check is also performed to ensure the consistency of judgments. The above process is repeated for each criterion. Finally, the candidate's overall score S_i measuring its overall performance with respect to all criteria is calculated using equation (7).

Table 5: Pair-wise Comparison Matrix for 4 Candidates with respect to E1

E_1	C_1	C_2	C_3	C_4
C_1	1	1/2	1/3	5
C_2	2	1	1/2	7
C_3	3	2	1	9
C_4	1/5	1/7	1/9	1

Table 6: Candidates ' Relative Scores with respect to E1

E_1	Relative Score	Label
C_1	0.174	S_{11}
C_2	0.293	S_{21}
C_3	0.489	S_{31}
C_4	0.044	S_{41}

$$S_i = f(S_{ij}, W_j) = \sum_{j=1}^{m} S_{ij} W_j, where: i = 1,...,l \quad (7)$$

4.2. Usage of BBN

The BBN technique uses a directed graph notation consisting of nodes and arcs to model the problem. The graph is used to visually illustrate the causal relationships between these nodes. Nodes represent uncertain variables while directed arcs reflect the causal relationship between them. Moreover, each node is associated with a probability table called node or conditional probability table (NPT or CPT respectively) to denote the causal influence [36].The process of using BBN includes the following two activities (see Figure 5).

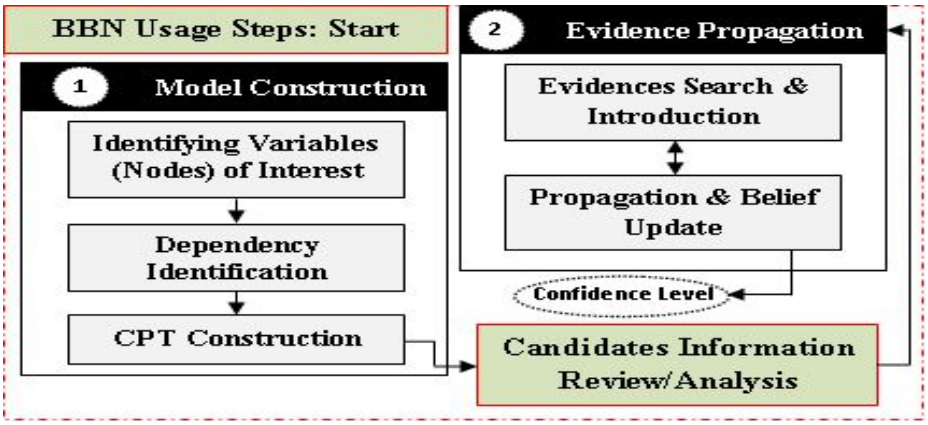

Figure 5: BBN Usage Process

4.2.1 Model Construction

Constructing a BBN model usually starts with identifying a set of uncertain variables representing the problem and the causal relationship (i.e. dependency) between them. In COTS selection, our concern is the explicit representation of how confident the evaluators are about scores S_i created by AHP and assigned to the

candidates. These scores are calculated using the relative scores S_{ij}. Furthermore, the criteria is associated with a set of attributes (e.g. performance criterion has response time, and throughput as attributes). Consequently, scores $S(A_{jk})$ that represent to what extent these attributes are satisfied by COTS candidates influence the relative scores S_{ij} because $S(A_{jk})$ is used to calculate S_{ij}. Moreover, how confident evaluators are about $S(A_{jk})$ influences how confident they are about S_{ij}.

Assuming that:

$Bel(C_i)$: candidate's confidence level refers to how confident evaluators are about scores (S_i)

$Bel(E_j)$: criterion's confidence level refers to how confident evaluators are about relative scores (S_{ij})

$Bel(A_{jk})$: attribute's confidence level refers to how confident evaluators are about scores $S(A_{jk})$

So,

$$Bel(C_i) \propto Bel(E_j) \propto Bel(A_{jk}) \quad \textbf{(8)}$$

Thus, it is required to determine:

$$Bel(C_i) \mid Bel(E_j) \mid Bel(A_{jk}) \quad \textbf{(9)}$$

This means, estimating $Bel(C_i)$ given $Bel(E_j)$which is calculated given $Bel(A_{jk})$. In summary, the BBN model for the COTS selection problem has three levels:

- The upper level contains nodes that represent COTS candidates.
- The intermediate level contains nodes that represent evaluation criteria.
- The lowest level contains nodes that represent attributes associated with evaluation criteria.

An abstract view for the BBN model of the COTS selection problem is shown in Figure 6.

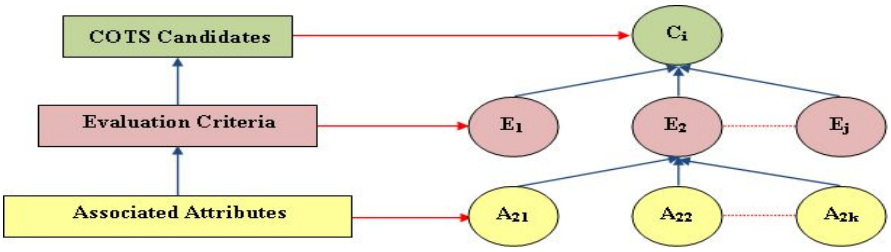

Figure 6: BBN Model for COTS Selection Problem

Considering Figure 6, the BBN model has a poly-tree structure (i.e. each node has multiple parents. For example attributes (A_{21}, A_{22}, and A_{2k}) are the parents of criterion E_2). Once the model structure is specified, a node probability table (NPT) is defined for each node in the model. The NPT represents the causal influence between the model nodes. Thus, the NPT associated with the candidate (C_i) node represents the conditional probability $P(C_i \mid E_1, E_2, ..., E_m)$ and the NPT associated with the evaluation criterion (E_j) represents the conditional probability $P(E_j \mid A_{j1}, A_{j2}, ..., A_{jn})$. Therefore, the NPT entries

specify the conditional probability of the node given the probability of its parents. Table 7 shows an example for the NPT. The NPT probability entries are specified by:

- Applying Bayes' theorem.
- Utilizing domain experts' knowledge.
- Using historical and empirical data about COTS products.

Table 7: NPT Example

$P(E_1)$	$P(E_2)$	$P(C_i)$	
		High	Low
Low	Low	0.15	0.85
Low	High	0.70	0.30
High	Low	0.90	0.10
High	High	0.95	0.05

4.2.2. Evidence Propagation

The evaluation team starts reviewing and analyzing information provided by COTS vendors and available through their websites, product documentation, demonstration, product prototypes, etc. The information review and analysis process searches for evidence and observations that increase or decrease the evaluation team's confidence level about scores assigned to the attributes, criteria, and candidates. Once new evidence is discovered, it is introduced into the BBN model and propagated to update the confidence in other nodes (i.e. evaluation criteria and COTS candidates). The message passing algorithm is selected for propagating evidence through the BBN model because it is suitable for Bayesian models with poly-tree structure. The message passing algorithm relies on the exchange of messages between nodes when new evidence is discovered. There are two types of messages: one **(π)** is for the forward (i.e. bottom-up or parent-to-child) propagation and the second **(λ)** is for the backward (i.e. top-down or child-to-parent) propagation. Once a node receives a π message from one of its parent nodes, it updates the belief or confidence associated with it, calculates (π), and sends it to its children. The confidence is calculated using equation (10).

$$Bel(C_i) = \alpha\lambda(Ci)\pi(C_i) \quad \textbf{(10)}$$

Where:

α: is a normalizing factor.

Considering the causal relationship in the COTS selection problem, only the forward propagation (i.e. the propagation of evidence from attributes nodes towards COTS candidate nodes) is performed. So, λ is initially set to 1 and is not changed during the evidence propagation process. However, π messages are calculated as follows:

$$\pi(x_i) = \sum_{u_1,\ldots,u_n} P(x_i \mid u_1,\ldots,u_n) \prod_{j=1,2,\ldots,n} \pi_x(u_j) \quad \textbf{(11)}$$

Where:

x_i: represents either a criterion or a candidate node.

u_j: represents x_i's parents.

For example, if x_i is a candidate node, u_j represents x_i's parents (i.e. evaluation criteria), and if x_i is a criterion node, u_j represents x_i's parents (i.e. associated attributes). In summary, when evidence that support scores assigned to an attribute or a criterion node, is discovered, the message (π) is sent from the attribute or the criterion node to its children (i.e. the evaluation criterion or candidate respectively) which in turn updates the belief or confidence associated with it, calculates (π) and then sends it to the COTS candidate node. The COTS candidate node updates the belief or confidence associated with it reflecting the confidence level about scores assigned to the candidate.

The process of introducing and propagating evidence and updating the confidence level in other model nodes continues until one of the following is reached:

- The evaluation team reaches a reasonable level of confidence.
- The evaluation team has reviewed and analyzed all available information about the candidates and no new information is discovered.
- Although, the evaluation team discovered new evidence, there is no change any more in the overall confidence level.
- Searching for more evidence leads to significant project cost and schedule overruns.

4.3. Selection Decision

In this step, evaluators use results produced by the AHP technique (i.e. candidates' scores $S(Ci)$) and the BBN technique (i.e. confidence $Bel(Ci)$). The utility of a candidate that reflect its performance with respect to the evaluation criteria and the evaluators' confidence level is defined as follows:

$$U(C_i) = S(C_i) * Bel(C_i) \quad \textbf{(12)}$$

Decision makers select the candidate with the maximum utility.

4.4. The UnHOS Support Tool

The UnHOS tool supports the evaluators and decision makers during the application of the UnHOS method. The UnHOS tool is implemented using visual C# and MS-Access. The tool components use Open Database Connectivity (ODBC) to access the database. Figure 7 shows the conceptual structure of the UnHOS tool and Figure 8 shows a screenshot of the GUI of UnHOS tool. As shown in Figure 7, the tool that supports the UnHOS consists of several components:

1. The graphical user interface (GUI) through which evaluators interact with the tool. The GUI facilitates and supports the interaction between evaluators and other tool components such as AHP and BBN modules.
2. The preparation component supports project-related tasks such as setting up a new project and its properties (e.g. project budget, and deadline), defining evaluation criteria and their attributes, handling information about COTS candidates and vendors, defining user roles creating user accounts, etc.
3. The AHP and BBN components are computational modules that deal with the functionality of AHP and BBN techniques including eliciting evaluator preferences towards the criteria and associated attributes, estimating relative and overall scores of candidates, introducing new evidence, and propagating changes.

4. The results and analysis component prepares data created by the AHP and BBN components and presents it to the evaluators (e.g., diagrams, bar charts, graphs, etc.). Further, it facilitates the COTS selection decision by consolidating the results created by the AHP and BBN components.

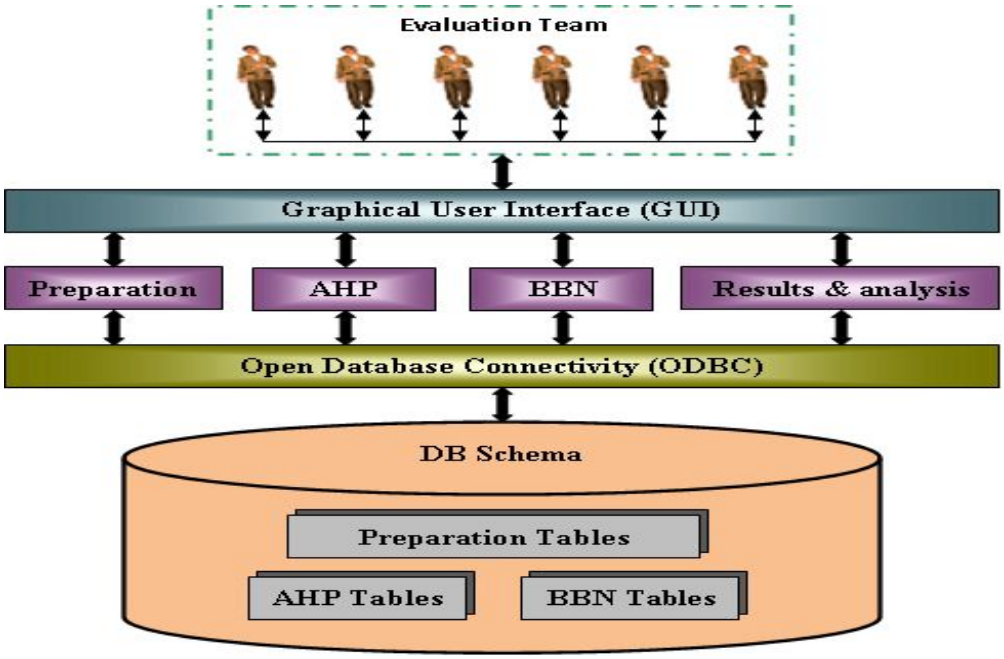

Figure 7: UnHOS Support Tool Architecture

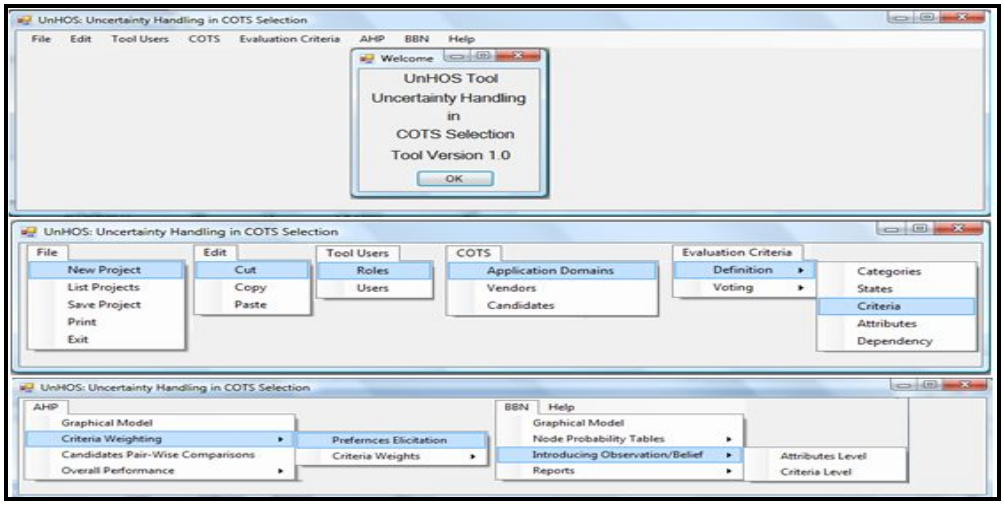

Figure 8: GUI Screenshot of the UnHOS Tool

5. The database schema component is used to store data that facilitates the functionalities of different components. Figure 9 shows the Entity-Relationship

Diagram for the UnHOS. The tables in the database schema are divided into three main categories:

a. The Preparation tables are used to store data about projects, criteria, candidates, vendors, evaluators.

b. The AHP tables are used to store data about evaluator preferences, scores, and any data related to the AHP functionality.

c. The BBN tables are used to store data related to the BBN functionality such as node probability tables, belief, evidence, etc.

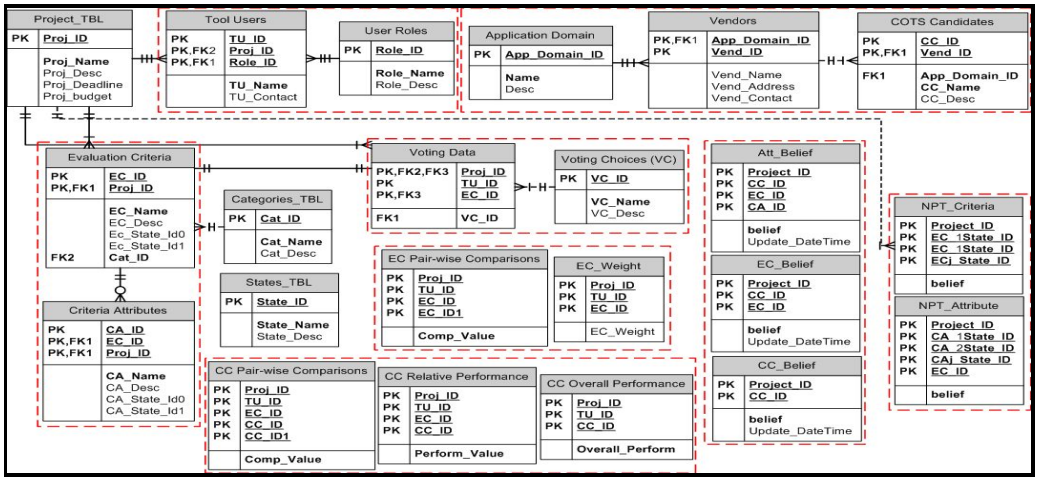

Figure 9: Detailed DB Schema: Tables and Relations

5. Case Study: Airline Reservations System (ARS)

5.1. ARS Overview

Databases are used to maintain internal records that support many business activities. With the rapid changes in computer technology and the decline of hardware costs, many applications are becoming database management systems. The power of databases comes from a body of knowledge and technology that has developed over several decades and is embodied in specialized software called database management systems (DBMS). A DBMS is a powerful tool for creating and managing large amounts of data efficiently and allowing it to persist over long periods of time. These systems are among the most complex types of software. In this case study, UnHOS and its tool are used to select the best DBMS candidate to develop an ARS. Examples of ARS requirements are:

1. The Airline Reservations System should include a set of queries that inquire for flights leaving around a certain time from one given city to another, available seats, and ticket prices. Another set of queries should allow changes to flight and reservation information.

2. The Airline Reservations System should allow concurrent access by several users while preventing mutual exclusiveness that could happen by assigning the same seat to more than one customer.

3. The Airline Reservations System should be able to house and manage the following sets of information:
   a) Information about reservation by a single customer on a single flight, such as the availability of a flight, service class, seat location, and meal preference.
   b) Information about the flight, such as the airport name that the customer will fly from and to, as well as the departure and arrival times.
   c) Information about the ticket price, discount, and taxes.
   d) Information about the visa requirements of the target destination, luggage permitted, and carry-on baggage allowance.

5.2 Evaluation Team

In the ARS case study, four evaluators (namely S1, S2, S3, and S4) participated in the process of acquiring a suitable DBMS to develop ARS. The four evaluators are information technology (IT) and database experts and academic researchers who are specialized in database systems at the University of Alberta, the University of Manitoba, and the University of California.

5.3. ARS: The Application of UnHOS

Figure 10 shows an extended version of the UnHOS method as was applied in the ARS case study. The figure contains the same activities described before in Figure 2, and extended by showing: (1) the output of each activity (shown by the ovals drawn in dotted lines), and (2) necessary roles to conduct each activity. Figure 10 will be used as a reference in this section. The application of UnHOS and its tool usually starts with recording detailed information about the project (e.g. name, budget, and deadline), evaluation team members (e.g. name, contact, domain experience) involved in the evaluation and selection process, the evaluation criteria and possible associated attributes, and COTS candidates and their vendors. Figure 11 shows a screenshot of a number of forms used to record this information. The following subsections discuss the application process in detail.

5.3.1 Applying UnHOS: Steps 1-4 of Evaluation Process

5.3.1.1 Step 1: Evaluation Criteria Definition

There are many criteria that can be considered when evaluating DBMS software packages. Most of the evaluation criteria presented in the literature are based on straightforward measurements. This includes the maximum table size, maximum database size, available data types, operating systems supported, number of processors supported, native connectivity and compatibility with other software, and the average price per client. However, such criteria do not match the unique feature of UnHOS method and tool in handling COTS problems that are associated with uncertainty. In view of that, the evaluators started by conducting a primarily comparative survey to review the specifications and features offered for most of the DBMS software available in the market. The participants decided to focus on the more subjective criteria that are important and challenging to many business process activities. The participants ended up by defining the evaluation criteria shown in Table 8. These dimensions (i.e., evaluation criteria) are not intended to be a complete list and are not presented in any particular order.

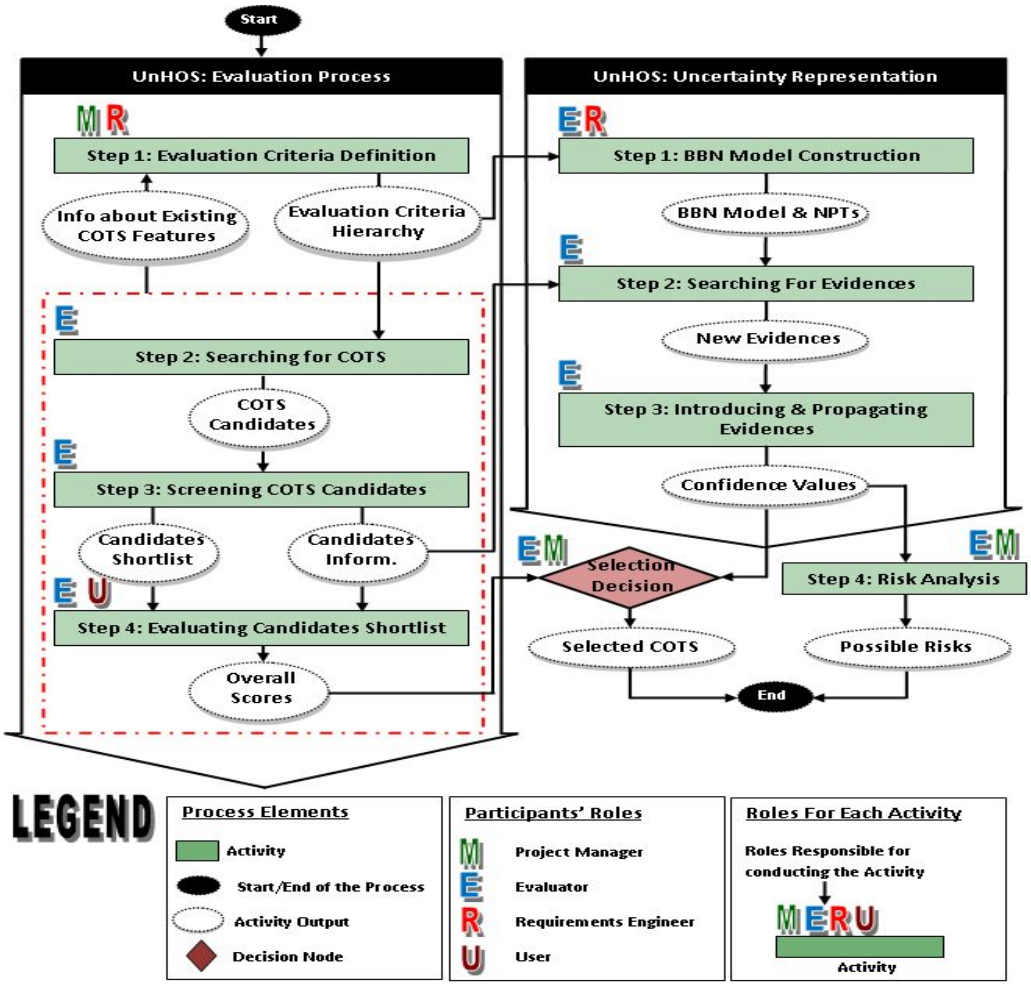

Figure 10: An Extended Version of UnHOS as Applied in DLS and ARS Case Studies

Table 8: Summary of Evaluation Criteria for ARS Case Study

ID	1	2	3	4	5	6
Name	Usability	Security	Functionality	Performance	Recoverability	Impact

Figure 11 shows a part of the hierarchical graph for the security criterion to give an idea about the graph structure in the ARS case study.

5.3.1.2 Steps 2 and 3: Searching for and Screening COTS Candidates

The evaluators search for DBMS and an initial list of DBMS candidates is identified based on the availability of the documentation that describes it, the familiarity of the database community with it, and its domain of applicability. Before the evaluation process can begin, operational and functional requirements that the DBMS software must satisfy should be determined.

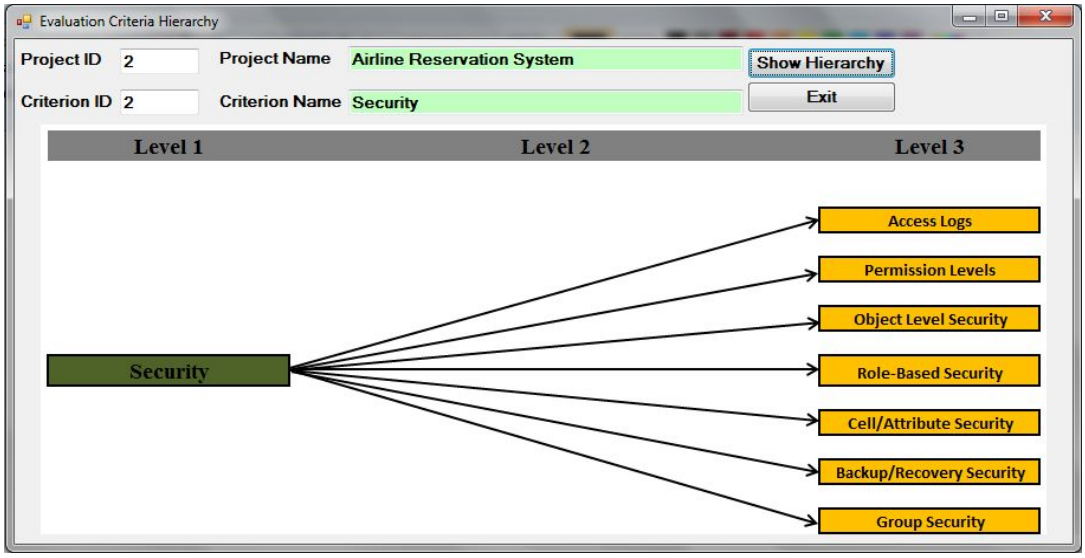

Figure 11: A Part of Hierarchical Criteria Graph in ARS

In particular, the evaluators should determine on what computer hardware and under what operating systems the software must run. The evaluators decided to use Microsoft Windows Server 2008 as an operating system to manage hardware and software resources of a more advanced 64-bit architecture. As a result, the initial list of DBMS candidates is screened to the following 3 potential DBMS software packages summarized in Table 9.

Table 9: Potential DBMS Candidates in for ARS Case Study

DBMS	Vendor	Architecture
SQL-Server 2008	Microsoft Corporation	64-Bit
Oracle 10*g* Release 2	Oracle Corporation	64-Bit
MySQL 5.1	Sun Microsystems Inc.	64-Bit

The evaluators were provided sufficient documentation about the ongoing study, clear instructions about the evaluation task, and equal amount of time to complete their evaluation using a well-prepared questionnaire.

5.3.1.3 Steps 4: Evaluating DBMS Candidates

AHP is used to evaluate the three DBMS candidates as follows:

5.3.1.3.1 Developing AHP Hierarchical Model for COTS Selection Problem

The evaluation criteria and short list of COTS candidates are used to develop an AHP hierarchy model for COTS selection in the ARS case study. The hierarchy AHP model has three levels. The first level is the main goal of the COTS selection problem which is selecting the best DBMS candidate. At the second level, there are a number of evaluation criteria (i.e. the criteria summarized in Table 8). Eventually, at the lowest level, there are the most promising DBMS candidates (see Table 9). There may be intermediate levels between level 2 and 3. These intermediate levels have strategic and technical criteria. Figure 12 shows the

AHP hierarchical model for the ARS case study. For simplicity, Figure 12 does not show the levels between levels 2 and 3.

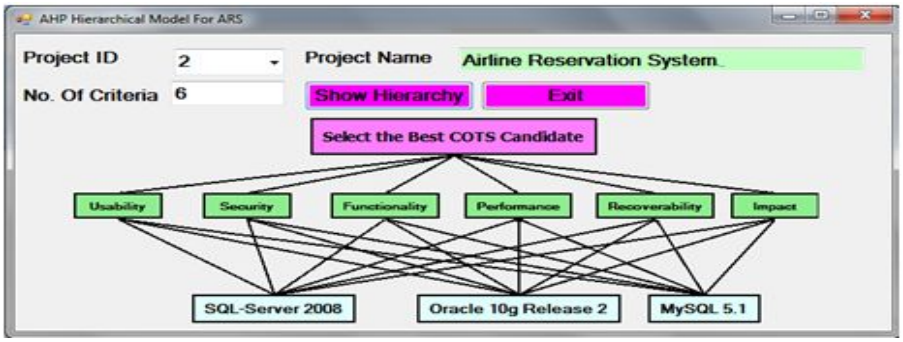

Figure 12: AHP Hierarchical Model for 6-Criteria in ARS Case Study

5.3.1.3.2 Step 2: Weighing of Evaluation Criteria

In this step, each evaluator develops his pair-wise comparison matrix that is used to calculate criteria weights. The interface shown in Figure 13 is used to complete this matrix. Figure 14 shows the criteria weights of the four evaluators.

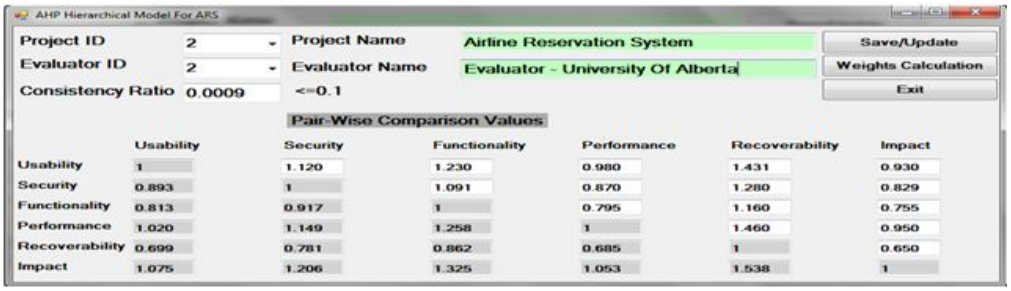

Figure 13: ARS: Preference Elicitation Interface for Evaluator 2

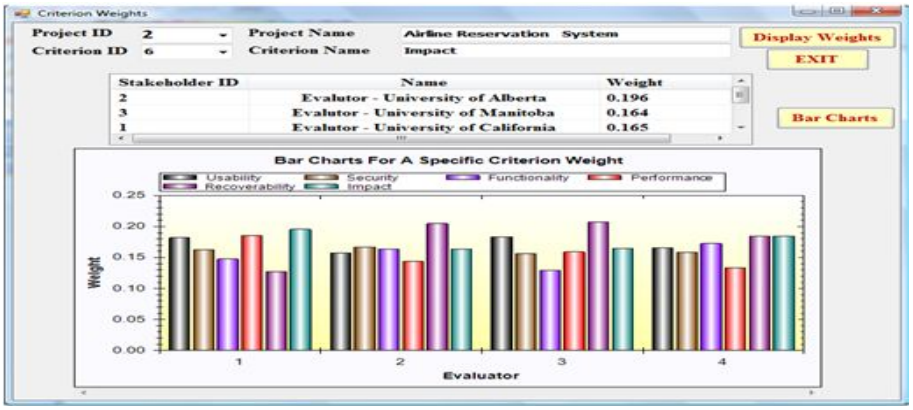

Figure 14: Criteria Weights for the Four Evaluators in ARS Case Study

Table 10 summarizes the calculated criteria weights of the four evaluators By plotting these data, as shown in Figure 15, we noticed that there are some extreme values (i.e., outliers) that fall well outside the overall pattern of the data and can potentially mislead the result of evaluation, and consequently the final selection decision.

Table 10: Summary of Criteria Weights for the Four Evaluators

Evaluator	Usability	Security	Functionality	Performance	Recoverability	Impact
1	0.184	0.156	0.129	0.159	0.207	0.165
2	0.182	0.162	0.148	0.186	0.127	0.196
3	0.158	0.167	0.164	0.143	0.205	0.164
4	0.166	0.158	0.173	0.134	0.185	0.185

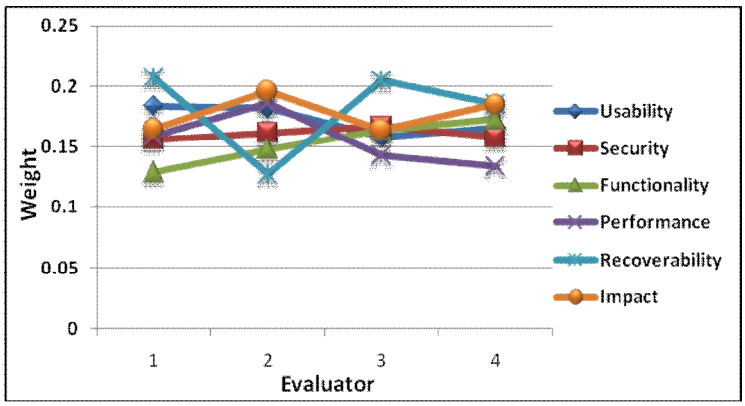

Figure 15: Scatter Plot of the Criteria Weights for the Four Evaluators

To overcome the effect of outliers, we decided to eliminate the upper and lower deciles and include only the middle 80% of the ranked data. We implemented this technique by computing the inter-percentile values that are within the upper 90 percentile and lower 10 percentile. The results are also given in Table 11.

Table 11: Normalized Results

Criteria	Usability	Security	Functionality	Performance	Recoverability	Impact
Middle 80% Inter-percentile	0.023	0.0089	0.0356	0.041	0.062	0.028
Normalized Inter-percentile	0.116	0.045	0.178	0.207	0.311	0.143

In summary, the normalized inter-percentile values given in Table 11 will be used as criteria weights to calculate overall scores of the DBMS candidates.

8.4.3.1.3.3 Step 3: Candidates Ranking

Each evaluator performs pair-wise comparisons between DBMS candidates with respect to each criterion. The interface used by the evaluator is shown in Figure 16. Figure 17 summarizes the ranking of the three DBMS candidates according to the four evaluators. As

shown in Figure 17, Oracle 10g Release 2 has the highest overall performance/score for all evaluators. Therefore, AHP results suggest selecting the Oracle DBMS.

5.3.2 Applying UnHOS: Uncertainty Representation

The process of using BBN to explicitly represent uncertainty during the evaluation process includes three steps (see Figure 10). These three steps are summarized below.

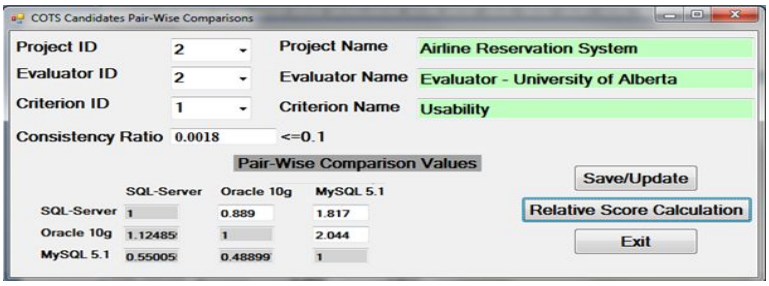

Figure 16: ARS: COTS Candidates Pair-Wise Comparisons

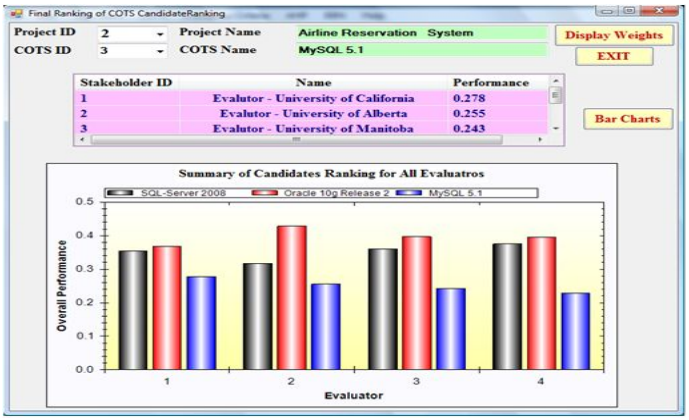

Figure 17: ARS: DBMS Candidates Ranking for the Four Evaluators

5.3.2.1 BBN Model Construction

In this step, the evaluation team performed the following tasks to develop a BBN model for the ARS case study (Figure 18 shows a part of the BBN model):

1. Identify variables of interest that represents nodes in BBN model. We identified three types of variables (see Figure 18):
   a. Variables representing attributes associated with the criteria (e.g. Automatic recovery, response time, etc.)
   b. Variables representing the evaluation criteria (e.g. Recoverability, Performance, etc.)
   c. Variables representing confidence in scores assigned to COTS candidates.
2. Identify the causal relationship between these variables (see equation 3.11)
3. Construct node probability tables (NPTs).

5.3.2.2 Steps 2-3: Searching for, Introducing, and Propagating Evidences

Each evaluator reviews and analyzes COTS candidates' information. The goal of the information review and analysis is to find evidence that increases or decreases the evaluation team's confidence level in the relative and overall scores. Figure 19 shows the interface used to introduce and propagate new evidence. When the evaluation team found new information that influences the confidence level that a criterion or an attribute is satisfied by a particular COTS candidate, they entered a value in the range from 0 to 1. Zero means that the evaluators have no confidence at all that the criterion or attribute is satisfied by the candidate. However, one means that they are completely confident. Values between 0 and 1 represent different levels of confidence. Then, the evidence propagation process starts to update confidence values in model nodes that are related to the node for which new evidence is discovered. As shown in Figure 19, the second evaluator provides a new confidence value (i.e. 0.65) for the response time attribute. Therefore, the evidence propagation process changes the confidence values in performance and SQL-Server 2008 candidate nodes. Figure 20 shows an example for the confidence summary for the second evaluator.

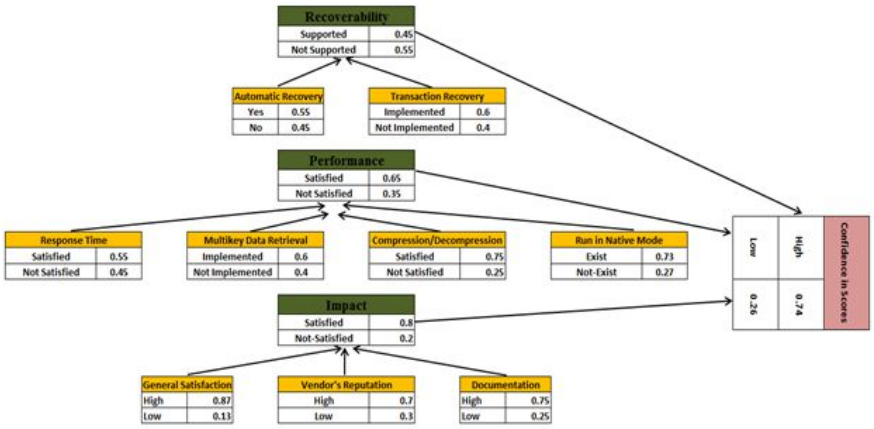

Figure 18: A Part of BBN Model for the ARS Case Study

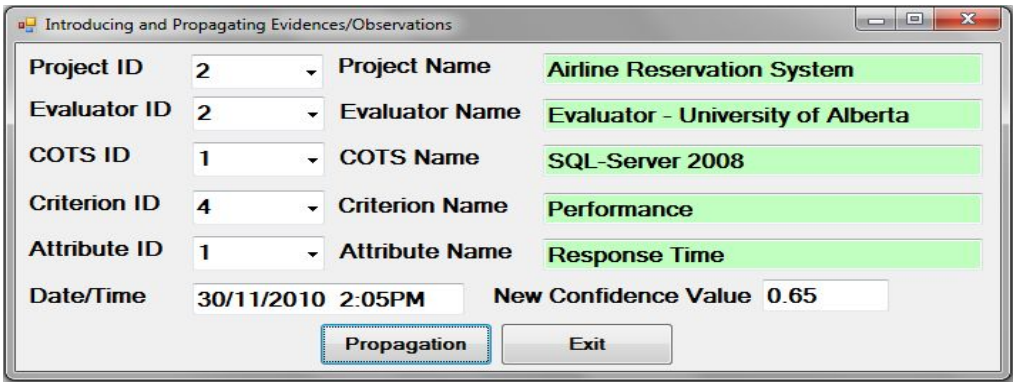

Figure 19: Interface for Introducing and Propagating Evidence

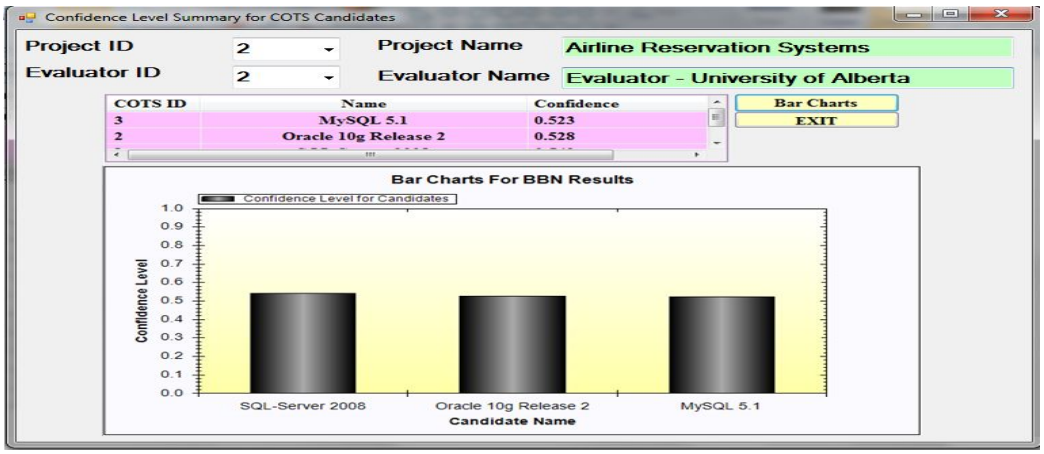

Figure 20: Evaluator 2: Confidence Level in Scores Assigned to DBMS Candidates

Table 12 summarizes scores calculated by AHP, confidence values produced by BBN, and utility calculated by equation 12 for the four evaluators. However, Figure 21 graphically represents the data shown in Table 12.

Table 12: Summary of Scores, Confidence Values, and Utility in ARS Case Study

Evaluator 1: University of California			
	Score	Confidence	Utility
SQL-Server 2008	0.355	0.512	0.182
Oracle 10g Release 2	0.367	0.517	0.189
MySQL 5.1	0.278	0.538	0.149
Evaluator 2: University of Alberta			
SQL-Server 2008	0.317	0.523	0.166
Oracle 10g Release 2	0.428	0.528	0.226
MySQL 5.1	0.255	0.538	0.137
Evaluator 3: University of Manitoba			
SQL-Server 2008	0.36	0.527	0.189
Oracle 10g Release 2	0.397	0.54	0.214
MySQL 5.1	0.243	0.551	0.134
Evaluator 4: University of California			
SQL-Server 2008	0.375	0.502	0.188
Oracle 10g Release 2	0.396	0.529	0.209
MySQL 5.1	0.229	0.535	0.123

The four evaluators defined three classes for the confidence level as follows:

High confidence: Evaluators are very sure that the information about the candidates is complete, accurate, and consistent as well as the scores assigned are correct.

Medium confidence: Evaluators are somewhat sure of the information and scores.

Low confidence: Evaluators are not sure of the information and scores.

Table 13 shows value ranges of confidence levels that participants of the case study used for low, medium, and high. These classes and value ranges may change from project to project. According to Tables 12 and 13, confidence level for the four evaluators is considered as medium confidence since it lies between 50% and 80%.

5.3.3 ARS: Selection Decision

As shown in Table 12 and Figure 21, the score values and utility suggest the selection of Oracle 10g Release 2 candidate since it has the highest score and the highest utility for all evaluators. Although, uncertainty representation does not influence the selection decision, it is still beneficial because for Oracle 10g Release 2, the average of evaluators' confidence values is 53% which is relatively low. This indicates that the evaluators are not confident about the score values and there is still a 47% chance of possible mismatch risk between the selected DBMS candidate and ARS requirements. It is also recommended that evaluators look for more information that increases their confidence level.

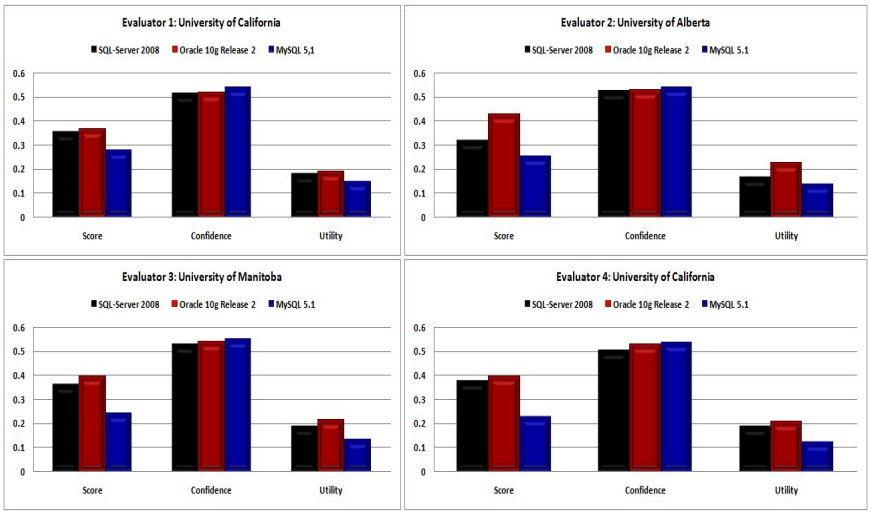

Figure 21: Summary of Scores, Confidence, and Utility of Candidates in ARS Case Study

Table 13: Confidence Values and Classes

Confidence Class	Value (V)
Low	$0 \le V < 50$
Medium	$50 \le V < 80$
High	$V \ge 80$

5.3.4 Discussion of the Results

The main benefit of using UnHOS to represent uncertainty during COTS evaluation is to identify possible mismatch risks that may be associated with the COTS selection decision. According to the ARS results, there are several scenarios related to possible risks associated with the COTS selection decision. These scenarios can be divided into two groups:

- **Perfect Scenario**

This scenario corresponds to cases in which the evaluator confidence is relatively high. In these cases, a high confidence means low likelihood of error in the assigned scores compared to those with medium or low confidence. In both case studies, there is no perfect scenario since all confidence values are between 50% and 80%.

- **Risky and Very Risky Scenarios**

Risky and very risky scenarios correspond to cases in which evaluator confidence is medium or low, respectively. In both scenarios, there are possible errors in the scores S_i assigned to the candidates. As a result, decision makers might select an inappropriate candidate resulting in higher mismatch between system requirements and the selected candidate's features. Even though the confidence level does not influence the candidates ranking, the explicit representation of uncertainty using evaluator confidence is still beneficial to highlight possible mismatch risks associated with the selection decision. As an example in the ARS case study, there is a 47% chance of mismatch when selecting Oracle 10g Release 2 since the average confidence level is 53%. One of the added values of using UnHOS method and its tool is that they can be used to identify the criteria for which the evaluators' confidence is low or medium. Figure 22 shows the fourth evaluator's confidence assigned to the relative scores of Oracle 10g Release 2 with respect to the six evaluation criteria.

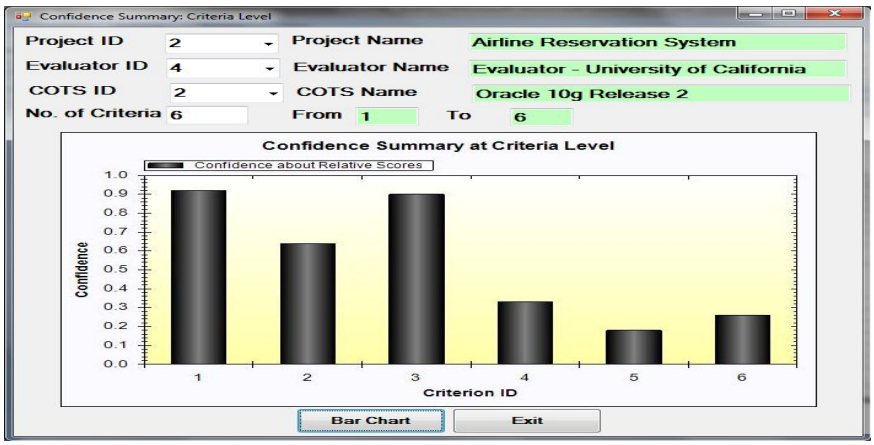

Figure 22: ARS: Confidence Summary at Criteria Level

As shown in Figure 22, Criteria 2, 4, 5, and 6 are considered as sources of risk since the corresponding confidence levels (i.e. 64%, 33%, 18%, and 26% respectively) are relatively low. These criteria need significant attention from the evaluators in particular if they are important (i.e. their weights are high) to stakeholders. Furthermore, the UnHOS method and its tool can be used to identify which attributes are sources of risk for each criterion. Figure 23 shows the attributes associated with criterion 4 (i.e. Performance) and corresponding confidence values assigned to Oracle 10g candidate.

As shown in Figure 23, attributes 2, 3, and 4 (i.e. number of simultaneous users, resources utilization, and throughput respectively) are considered as sources of risk since the corresponding confidence levels (i.e. 55%, 39%, and 25% respectively) are relatively low. These attributes also need significant attention from the evaluators.

In conclusion, it is important to consider the influence of confidence levels on the final COTS selection decision. The extent to which the confidence influences the outcome depends on the scores, confidence levels, difference between these scores, and difference between the

confidence levels. Although, the explicit representation of uncertainty may not influence the final COTS selection decision, it is beneficial since it can be used to determine which criteria and attributes lead to a low confidence level.

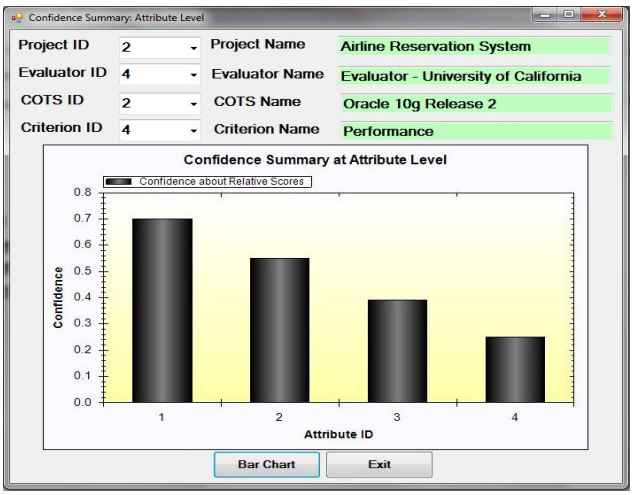

Figure 23: ARS: Confidence Summary at Attribute Level

6. Conclusions and Future Work

Uncertainty is a major challenge associated with COTS evaluation and selection. To handle uncertainty inherent to information related to COTS candidates and vendors, we proposed the UnHOS method to evaluate COTS candidates while explicitly representing uncertainty. UnHOS uses the BBN technique to explicitly represent uncertainty in terms of evaluator confidence about AHP scores. Furthermore, we presented a tool that supports evaluators and decision makers when using UnHOS. The use of the UnHOS method and its tool was illustrated with the help of an airline reservation system example. This paper makes several contributions:

- The UnHOS method, in contrast to other COTS selection methods, considers two factors which are the scores assigned to the candidates and the evaluators' confidence about these scores, to select the best candidate. The utility of each candidate is calculated using these two factors and the candidate with the highest utility will be selected.
- We have demonstrated the use of UnHOS in selecting the best off-the-shelf DBMS for an airline reservation system.
- In the ARS case study, considering uncertainty in terms of confidence does not influence the final COTS selection decision. However, it may influence the COTS selection decision in other CBD projects. To what extent the decision is influenced depends on:
    - Whether the confidence is low, medium, or high and the difference between these confidence levels.
    - The AHP scores and the difference between them.

- When the confidence level is high, the possibility of having errors in the scores S_i is low and most probably uncertainty will not influence the COTS selection decision. However, when the confidence level is low or medium, the possibility of having errors is high. This requires special attention by the evaluators to avoid selecting an inappropriate candidate or ignoring a potentially good one.
- Although, the explicit representation of uncertainty may not influence the selection decision (e.g. in the case of having a high confidence level), it is still beneficial to identify possible sources of mismatch risk. For example, if the average confidence about scores assigned to Oracle 10g is 53%, the 47% represents the risk of having a mismatch between the system requirements and Oracle 10g features.
- The UnHOS method can be used to determine which criteria and attributes lead to a low confidence level and in turn evaluators can ask vendors for information about these criteria and attributes, demo, or hand-trying.

Future research that is desirable includes:

1. Improving the scalability of UnHOS so that it can be used with a large number of criteria, attributes, and candidates and the overall evaluation time, effort, and cost is still reasonable.
2. It is necessary to use UnHOS and its tool in several case studies to generalize the obtained results.
3. Completing the implementation of the UnHOS tool to include modules related to scalability and improving its maturity by using it in several case studies and getting feedback from its users.

References

[1] Morisio M., Seaman C.B., Basili V.R., Parra A.T., Kraft S.E., Condon S.E., COTS-based software development: Processes and open issues, Journal of Systems and Software, Volume 61 , Issue 3 (April 2002), pp. 189-199.

[2] Clark J., Clarke C., De Panfilis S., Granatella G., Predonzani P., Sillitti A., Succi G., Vernazza T., Selecting components in large COTS repositories, Journal of Systems and Software, Volume 73 , Issue 2 (October 2004), pp. 323-331.

[3] Oberndorf P., Facilitating Component-Based Software Engineering: COTS and Open Systems, In Proc. of the 5th Int. Symposium on Assessment of Software Tools –"SAST'97", June 1997, pp. 143-148.

[4] Abts C., Boehm B. W., Clark E. B., COCOTS: a COTS software integration cost model model overview and preliminary data findings, In Proc. Of The 11th ESCOM conference Munich, Germany, 2000, pp. 325-333.

[5] Boehm B. W., Port D., Yang Y., and Bhuta J., Not All CBS Are Created Equally: COTS-Intensive Project Types, LNCS Vol. 2580/2003, Springer Berlin/Heidelberg 2003, pp. 36-50.

[6] Ibrahim H., Far B. H., Eberlein A., On Agent-Oriented Requirements Engineering for COTS-Based Software Development (CBSD), In Proc. of the 20th Canadian Conference on Electrical and Computer Engineering (CCECE 2007), Vancouver, April 2007, pp.690-693.

[7] Mohamed A., Ruhe G., Eberlein A., COTS Selection: Past, Present, and Future, In Proc. Of 14th Annual IEEE Int. Conf. and Workshops on the Engineering of Computer-Based Systems (ECBS'07), 2007,pp. 103-114.

[8] Crnkovic I., Larsson M., Challenges of component-based development, Journal of Systems and Software,Volume 61 , Issue 3 (April 2002), pp. 201 – 212.

[9] Maiden N.A., Ncube C., Acquiring COTS Software Selection Requirements, IEEE Software, vol. 15 (2), 1998, pp. 46-56.

[10] Kontio J., A Case Study in Applying a Systematic Method for COTS Selection, In Proc. of the 18th International Conference on Software Engineering (ICSE'96), Berlin, Germany, 1996, pp. 201–209.

[11] Jeanrenaud J., Romanazzi P., Software Product Evaluation: A Methodological Approach, in: Software Quality Management II: Building Software into Quality, 1994, pp. 55–69.

[12] Tran V., Liu D. B., A risk mitigating model for the development of reliable and maintainable large-scale commercial-off-the-shelf integrated software systems, In Proc. of the 1997 Annual Reliability and Maintainability Symposium (RAMS), 1997, pp. 361–367.

[13] Ochs M., Pfahl D., et al., A method for efficient measurement-based COTS assessment and selection – method description and evaluation results, In Proc. of the 7th Int. Symposium on Software Metrics, IEEE Computer Society, London, England, 2001, pp. 285–294.

[14] Alves C., Castro J., CRE: A systematic method for COTS components selection, In XV Brazilian Symposium on Software Engineering (SBES), Rio de Janeiro, Brazil, 2001.

[15] Phillips B. C., Polen S. M., Add decision analysis to your COTS selection process, in: CroosTalk – The Journal of Defence Software Engineering, 2002.

[16] Boehm B. W., Port D., et al., Composable process elements for developing COTS-based applications, In Proc. of the 2003 Int. Symposium on Empirical Software Engineering, ISESE' 2003, ACM-IEEE, 2003, pp. 8–17.

[17] Kunda D., Brooks L., Applying social-technical approach for COTS selection, In Proc. of the 4th UKAIS Conference, University of York, England, 1999.

[18] Gregor S., Hutson J., Oresky C., Storyboard process to assist in requirements verification and adaptation to capabilities inherent in COTS, In Proc. Of the Int. Conference on COTS-Based Software Systems (ICCBSS-2002), Springer-Verlag, Florida, USA, 2002, pp. 132–141.

[19] Burgués X., Estay C., et al., Combined selection of COTS components, In: Proc. of the Int. Conference on COTS-Based Software Systems (ICCBSS-2002), Springer-Verlag, Florida, USA, 2002, pp. 54–64.

[20] Dorda S. C., Dean J., et al., A process for COTS software product evaluation, In Proc. of the 2nd International Conference on COTS-Based Software Systems (ICCBSS-2002), Springer-Verlag, Florida, USA, 2002, pp. 86–92.

[21] Chung L., Cooper K., Defining Goals in a COTS-Aware Requirements Engineering Approach, *Systems Engineering journal,* vol. 7 (1), 2004, pp. 61-83.

[22] National Research Council (US), Risk analysis and Uncertainty in Flood Reduction Studies, National Academic Press, 2000.

[23] Gashi I., Popov P., Stankovic V., Uncertainty explicit assessment of off-the-shelf software: A Bayesian approach Source, Journal of Information and Software Technology, Volume 51 , Issue 2 (February 2009), pp. 497-511

[24] Mead W. J., Uncertainty in Decision Making, Encyclopedia of Public Administration and Public Policy, Second Edition, 2008.

[25] Carney D.J.; Wallnau K. C., A basis for evaluation of commercial software, Journal Information and Software Technology, Volume 40, Issue 14, December 1998 , pp. 851-860.

[26] Adams R. J., Eslinger S., Best Practices for the Acquisition of COTS-Based Systems: Lessons Learned from the Space System Domain, LNCS Volume 2959/2004, 2004, pp. 203-205.

[27] Yu Y. Y., Johnson B. W., A BBN Approach to Certifying the reliability of COTS software systems, In Proc. Annual Reliability and Maintainability Symposium, 2003, pp. 19-24.

[28] Erdogmus H., Management of License Cost Uncertainty in COTS-Based Software Development: A Real Options Approach, Available online at http://citeseer.ist.psu.edu/erdogmus01management.html, 2001.

[29] Littlewood B., P. Popov, Strigini L., Assessment of the Reliability of Fault-Tolerant Software: a Bayesian Approach. In Proc. Of 19th Int. Conf. on Computer Safety, Reliability and Security, SAFECOMP'2000. 2000. Rotterdam, the Netherlands: Springer, pp. 294-308.

[30] Ibrahim H., Wanyama T., Eberlein A., Far B., Conflict Analysis in Commercial Off-The-Shelf (COTS) Based Development, accepted at the 22nd International Conference on Software Engineering and Knowledge Engineering (SEKE 2010), July 1-3 2010, San Francisco Bay, USA.

[31] Wanyama T., Far B. H., Negotiation Coalitions in Group-Choice Multi-Agent Systems, In Proc. of the 5th Int. Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS'06),Hakodate, Japan, pp. 408-410.

[32] Saaty T. L. , Decision Making for Leaders, Third edition, RWS Publications, 2001.

[34] Chin-Feng Fan, Yuan-Chang Yu, BBN-based software project risk management, Journal of Systems and Software, Volume 73 , Issue 2 (October 2004), pp. 193-203.

[35] Ibrahim H., Far B., Eberlein A. (2008) Towards Uncertainty Management in COTS-Based Development, In Proc. of the 12th IASTED Int. Conference on Software Engineering and Applications (SEA 2008), November 16–18, 2008, Orlando, Florida, USA.

[36] Ibrahim H., Far B. H., Eberlein A., Scalability Improvement in Software Evaluation Methodologies, In Proc. of the 10th IEEE Int. Conference on Information Reuse & Integration (IRI 2009), Las Vegas, USA, 2009, pp. 236-241.